Department of Computer Science and Engineering
The University of Texas at Arlington

System Test Plan

Team: Ground Control

Project: Sherpa Drone

Team Members:
Justin Crist
Matthew Leonard
Frank Robinson
Dakota Slay
Aaron Windham

Last Updated: April 8, 2015 10:55 AM

Table of Contents

1. Introduction....1
1.1 Product Concept....1
1.2 Product Scope....1
1.3 Testing Scope....2
2. References....3
2.1 System Requirements Specification....3
2.2 Architecture Design Specification....11
2.3 Detailed Design Specification....16
3. Test Items....19
3.1 Hardware Tests....21
3.2 Unit Tests....23
3.3 Component Tests....28
3.4 Integration Tests....31
3.5 System Validation....33
4. Risks....36
4.1 Risks Table....36
5. Features To Be Tested....37
5.1 Customer Requirements....37
5.2 Packaging Requirements....39
5.3 Performance Requirements....40
5.4 Safety Requirements....42

5.5 Maintenance and Support Requirements 43
5.6 Other Requirements 44
6. Features Not To Be Tested 45
6.1 Customer Requirements 45
6.2 Packaging Requirements 46
6.3 Performance Requirements 47
6.4 Safety Requirements 47
6.5 Maintenance and Support Requirements 47
6.6 Other Requirements 48
7. Testing Approaches 49
7.1 Strategy 49
7.2 Tools 49
7.3 Core Functionality 49
7.4 Test Metrics 50
8. Item Pass/Fail Criteria 51
8.1 Hardware Tests 51
8.2 Unit Tests 52
8.3 Component Tests 56
8.4 Integration Tests 59
8.5 System Validation 60
9. Test Deliverables 62
9.1 System Test Plan 62
9.2 Test Cases 62
9.3 Test Case Results 62
9.4 Defects 63
10. Test Schedule 64

10.1 MS Project Plan -- System Test Plan Phase 64

11. Approval 66

Document Revision History

Revision Number	Revision Date	Description	Rationale
0.1	3/23/2015	Initial version for Rough Draft	n/a
1.0	3/30/2015	Review version	Added breakdown diagram and cleaned up document
2.0	4/8/2015	Baseline version	Made changes based on review feedback

List of Figures

Figure #	Title	Page #
2-1	Architecture Layer Overview	11
2-2	Module Decomposition Chart	16
2-3	Producer-Consumer Matrix	17
3-1	Testing Composition Diagram	20

List of Tables

Table #	Title	Page #
2-1	Customer Requirements	3-5
2-2	Packaging Requirements	5-6
2-3	Performance Requirements	6-7
2-4	Safety Requirements	8
2-5	Maintenance and Support Requirements	9-10
2-6	Other Requirements	10
2-7	Data Flow Definitions	11-13
2-8	Requirements Traceability Matrix	18
3-1	Hardware Tests	21-22
3-2	Unit Tests	23-27
3-3	Component Tests	28-30
3-4	Integration Tests	31-33
3-5	System Validation	33-35
4-1	Risks Table	36
7-1	Metrics by Priority	50
8-1	Hardware Tests	51-52
8-2	Unit Tests	52-56
8-3	Component Tests	56-59
8-4	Integration Tests	59-60
8-5	System Validation	60-61
10-1	System Test Phase Schedule	64-65

1. Introduction

The System Test Plan will constitute the complete process for testing the Sherpa Drone system. This document will cover the different types of testing that will be used to ensure the system meets specifications. It will also detail the different tests that will be performed to ensure the system meets the requirements and fulfills the design as outlined in the Architecture Design Specification and Detailed Design Specification.

1.1 Product Concept

The purpose of the Sherpa Drone is automated guidance through a retail or warehouse environment. Users will interact with the Sherpa Drone through a touch screen kiosk. They will specify their product of interest at the kiosk through the search interfaces. The drone will launch from a base station adjacent to the kiosk and fly through the store, guiding the user to the location of their item. Once the drone has completed its mission, it will return to the base station. If more users wish to use the system, the drone will launch again. Otherwise, it will charge while waiting on interaction.

1.2 Product Scope

The Sherpa Drone will consist of a drone, base station, kiosk, database, and infrared receivers. The kiosk will be constructed using a raspberry pi and a touch screen monitor. The touch screen will be the exclusive means of interaction for both users and administrators. Customers will search for items they wish to locate via its interface. Administrators will use their own interface to maintain and update information about the store and products. All data will tie into the database. The database can be broken up into two different parts, the store layout and items database. The store layout will contain the location of aisles, display tables, and obstacles, such as pillars, that the system will need to know to be able to navigate the store. The items database will contain all the items in the store, and their locations.

The base station will provide a place to house the drone, charge the drone, and transmit signals. The base station will also handle calculating the path to the items and sending instructions to the drone on navigating the store. For the initial prototype, the kiosk and base station will be merged together. However, they do not need to be a single unit.

The drone will be a small aerial drone with an LED light and speakers so customers are able to keep track of the drone. Infrared receivers will be installed throughout the store. These receivers will determine the location of the drone as it moves in the store. The drone will also be armed with proximity sensors and an inertial measurement unit (IMU) to assist with navigation.

1.3 Testing Scope

Testing of the Sherpa Drone will not take place inside of a live store environment. Instead the system shall be tested in the UT Arlington CSE Senior Design Lab. The team will set up the layout of three cubicles in the lab inside of the kiosk, adding several locations who will be marked with tape inside the cubicles. The infrared sensors will be placed up on the lighting fixtures in order to create the RPS. The team will then run any of the tests that require the drone to be launched from within the lab which should be a safe environment if any issues do occur. The team will take every precaution to ensure that no property of the university or other teams is damaged. While this will not be a live store environment it does contain aisles and blocked spaces that should make it a good substitute for a real store.

2. References

This section provides references to the other documentation that was used in creating the System Test Plan. The documents referenced here are the System Requirements Specification (SRS), Architecture Design Specification (ADS), and Detailed Design Specification (DDS). The following subsections will contain the key information from each of these documents respectively.

2.1 System Requirements Specification

2.1.1 Customer Requirements

Table 2-1 Customer Requirements

SRS #	Requirement	Description	Priority
3.1	Search By Name	The system shall allow a customer to type in the name of an item to search for at the kiosk.	1 - Critical
3.2	Auto-Fill Search Bar	The system shall suggest possible items to the customer as the customer types into the search bar at the kiosk.	3 - Moderate
3.3	Search By Category	The system shall allow a customer to select an item from the search list based on category at the kiosk. This will provide them with a list of all items in the store in that category that they can browse to select.	2 – High
3.4	Follow Drone	The system shall allow a customer to follow a drone around the store towards the desired item location. The drone must fly at a reasonable pace and travel a path that the customer can follow without difficulty.	1 – Critical

3.5	Stay with Customer	The drone will match pace with the customer, slowing as necessary. If the customer falls behind, the drone must stop to wait. The drone must wait for the customer until the customer approaches again or thirty seconds have passed when the customer will be informed to start moving and if an additional fifteen seconds pass the system will cancel the search.	5 – Future
3.6	Open Administrator Console	The system shall allow an administrator to log in, with a password, to a special menu at the kiosk in order to add, remove, and move items in the store.	1 – Critical
3.7	Manage Items in Inventory	The system shall allow an administrator to add and modify items to the store with their name, category, and location at the kiosk. This will include adding a new category if none exists that works for this item.	1 – Critical
3.8	Change Store Layout	The system shall allow an administrator to change the layout of aisles and shelves within the store at the kiosk.	4 – Low
3.9	Track Inventory Changes	The system shall allow an administrator to update the locations of items automatically using RFID or other inventory tracking technology.	5 – Future
3.10	Make Drone Visible/Audible	The drone will use an LED light and sound cues to announce its activity to surrounding customers. The LED will support multiple colors so that if the drone is helping a customer and another approaches the kiosk and searches for an item it will inform them that, when the drone will lead them, it will have a different color.	2 – High
3.11	Mobile App Access	The system shall include a mobile app that allows customers to request the service of a drone while within the store.	5 – Future

3.12	Service Drone	The system shall allow an administrator or service personnel to perform maintenance on the drone, including charging the battery.	1 – Critical
3.13	Drone Automated Charge	The system will automatically charge the drone while it is not currently helping customers so that it will consistently have a full or near-full battery.	3 – Moderate

2.1.2 Packaging Requirements

Table 2-2 Packaging Requirements

SRS #	Requirement	Description	Priority
4.1	Kiosk Packaging	The kiosk will be enclosed in a soft edged metal box with a 7 inch touch screen on the front, a flat landing zone on top, and a power connection in the back.	2 – High
4.2	Kiosk Circuitry	Kiosk enclosure will house all required circuitry, wires and components out of sight from the customer.	4 – Low
4.3	Kiosk Power Source	Kiosk will utilize a single external power cable. This cable will provide the correct DC power to components from an AC plug.	5 – Future
4.4	Drone Packaging	The Drone will be a quad-copter style drone consisting of four rotors, a small frame, an RGB LED, a small speaker, a Styrofoam cover, and a central control board.	2 – High
4.5	Drone Circuitry	Drone circuitry, including the control board, will be housed entirely out of sight of the customer.	4 – Low
4.6	Drone Wires	Drone will enclose most wires from customer vision. The only acceptable exposed wires are those to the motors and from the batteries.	3 – Moderate

4.7	Drone Sensors	The drone will include ultrasonic sensors for the detection of objects in its vicinity.	2 – High
4.8	System Installation	The system designers will install the necessary product pieces at the retail environment.	2 – High

2.1.3 Performance Requirements

Table 2-3 Performance Requirements

SRS #	Requirement	Description	Priority
5.1	Drone Response Time	The drone will demonstrate responsive behavior through a light that turns on to indicate a message received. Messages from the base station to the drone will be delivered in an interval of under a second to allow prompt drone response.	1 – Critical
5.2	Drone Wake Time	The system will respond to a new item selection and wake the drone from a dormant state on the base station within fifteen seconds. Total time to calculate route, communicate with the drone, and begin travel must fall within this time.	3 – Moderate
5.3	Drone Travel Time	The drone will complete its travel in a reasonable time span as appropriate to the size of the store and the maximum number of items allowed in a single trip.	1 – Critical
5.4	Return After Completion of Guidance	The drone will return to the base station once travel has been executed for the customer. The drone will not be expected to wait on customer input. It may follow a path that is not appropriate to customers.	2 – High
5.5	Base Landing Upon Completion	The drone will execute a landing on the base station when guidance is complete. If the drone loses connectivity with the base station it will land in a designated safe zone in the store.	1 – Critical

5.6	Base Station Charging	The drone will execute a landing on the base station in the correct orientation to charge. If unable to charge, it will attempt a second landing. If this landing fails, the drone will land on the base station and notify the administrator.	3 – Moderate
5.7	Max Items Per Trip	The system will require that users specify no more than an allotted number of items per guidance operation. This number will be set by the administrator as part of system set up. Customers who wish to follow the drone to more items must return to the base station and complete another request. This limit will prevent a single customer request from monopolizing the use of the drone.	3 – Moderate
5.8	Minimum Operational Time	The drone will be able to maintain operational flight time of at least thirty minutes from a fully charged state.	2 – High
5.9	Single Drones Supported	The system will support the full operation of a single drone at a time. This drone will be responsible for guiding all customers.	1 – Critical
5.10	Operational Range	The drone will maintain operational contact with the base station at all times through a wireless signal. Any drone that loses contact with the base station through exceeding the operational range or interference will immediately land and power down in the designated safe zone.	2 – High

2.1.4 Safety Requirements

Table 2-4 Safety Requirements

SRS #	Requirement	Description	Priority
6.1	Drone Rotors	The drone rotors shall not harm a person or damage property.	1 – Critical
6.2	Drone Weight	The system shall be light enough to avoid damage to person or property if it were to fail.	1 – Critical
6.3	Property Damage	The system shall be designed so that it does not cause property damage. A collision with another object does not cause property damage.	1 – Critical
6.4	Eye and Ear Protection	The system shall not cause damage to peoples' eyes through its LED or to peoples' ears through the speaker.	1 – Critical
6.5	Drone Avoids Collisions	The drone shall fly in such a manner to avoid pedestrians or objects.	1 – Critical
6.6	Altitude Control	The system shall control its flight altitude to avoid hazards to customers or the environment. The flight altitude shall be set in the initial configuration of the product for a store. At this time the altitude shall be limited to a single level, i.e. a store with two floors would require separate drones for each floor.	1 – Critical
6.7	Down Draft	The system shall be designed so that rotor down draft does not cause damage or harm to person or property.	1 – Critical
6.8	Damage Avoidance	The system shall be designed to fail safely. Damaged parts shall not pose an immediate hazard to person or property.	1 – Critical

2.1.5 Maintenance and Support Requirements

Table 2-5 Maintenance and Support Requirements

SRS #	Requirement	Description	Priority
7.1	Source Code and Documentation Availability	All source code developed by Ground Control will be made available to the maintenance team and future development teams. This source code will include adequate comments to document its functionality. The team will make available all schematics and technical documentation, including: System Requirements Specification, Architectural Design Specifications, Detailed Design Specifications, and System Test Plan.	3 - Moderate
7.2	User Manual	The User Manual will contain instructions for operation of the system. It will instruct administrators in maintaining an accurate layout of their store and inventory of the items in the store. Another section will instruct customers on how to use the kiosk, which will be accessible from the kiosk itself.	3 – Moderate
7.3	Drone Maintenance	Standard maintenance of the drone, such as checking for damaged parts, will be included in the User Manual. The Ground Control team will train technicians to take over maintaining the system after completion of the project. If a software update is needed for the drone, the trained technicians will complete installation.	4 – Low

7.4	Kiosk Maintenance	Standard maintenance of kiosk will be included in User Manual. The Ground Control team will train technicians to take over maintaining the system after completion of the project. For software updates, administrators will be able to download the update onto a flash drive and install it on the kiosk.	4 – Low
7.5	Troubleshooting Guide	The troubleshooting guide will include solutions to commonly detected problems discovered during system testing and from users of the Sherpa Drone.	4 – Low

2.1.6 Other Requirements

Table 2-6 Other Requirements

SRS #	Requirement	Description	Priority
8.1	Configuration of Store	The system will allow the installation team to input the layout of the store into its memory. This will include where the aisles and other locations (tables, racks, etc.) where items are located in the store.	2 – High
8.2	Data Storage	The system will store inventory information in a database. This database will include the location data necessary for drone path-finding. It will also include category of each item to assist in user searches.	1 – Critical
8.3	Protected Administrator Console	The system will hide the administrator console from access by the customer by using a special code or sequence to bring up the log in page.	5 – Future

2.2 Architecture Design Specification

2.2.1 Architecture Design Diagram

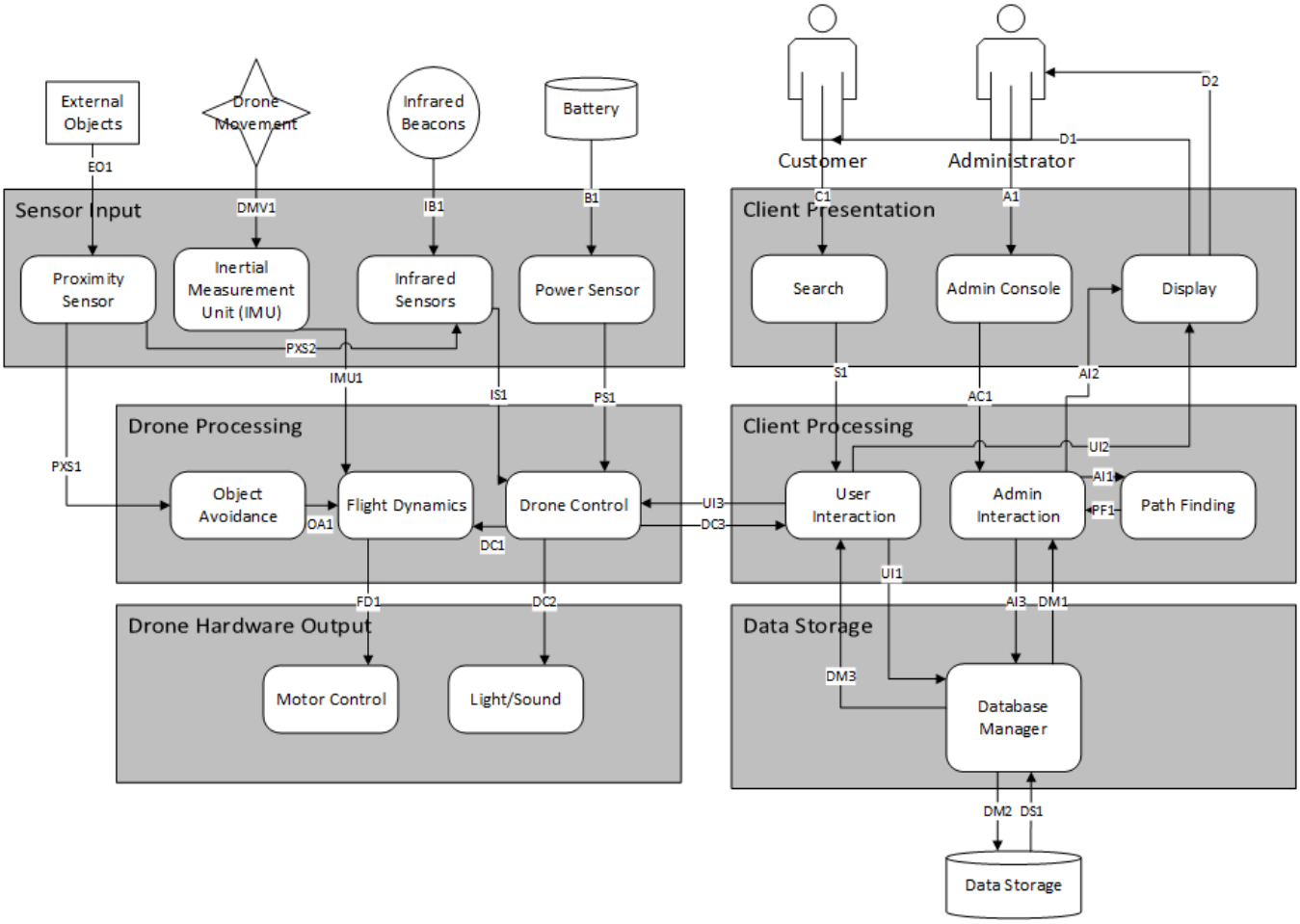

Figure 2-1 Architectural Layer Overview

2.2.2 Data Flow Definitions

Table 2-7 Data Flow Definitions

Element	Description
EO1	The presence of objects external to the system result in surfaces that reflect ultrasonic sound waves back to the Proximity Sensors resulting in a reading.
DMV1	The physical movement seen as accelerations of the drone are detected by the Inertial Measurement Unit (IMU).

IB1	The beacon mounted on the drone is observed by the two infrared sensors in the room and the beacon on the base station is observed by the drone mounted sensor.
B1	The battery's voltage and current are detected by the Power Sensor.
C1	The customer taps on the console's touch screen to send signals to the Search GUI objects.
A1	The administrator taps on the console's touch screen to send signals to the Admin Console GUI objects.
PXS1	The Proximity Sensors produce a structure of data containing the integer distance to any objects in inches both ahead of and underneath the drone and sends this to the Object Avoidance subsystem.
PXS2	The Proximity Sensors send the distance to the ceiling to the Infrared Sensors subsystem to give it the Z position of the drone.
IMU1	The IMU produces a structure of data containing the float accelerations in the X, Y, and Z axes to the Flight Dynamics subsystem.
IS1	The Infrared Sensors produce a structure of data containing the integer X, Y, & Z position values representing the current location of the drone to the Drone Control subsystem.
PS1	The Power Sensor sends an integer value of the remaining battery percentage to the Drone Control subsystem.
S1	The Search subsystem sends string and integer values representing customer selections to the User Interaction subsystem to process based on the called method.
AC1	The Admin Console subsystem sends string, integer, and Item object values representing administrator selections to the Admin Interaction subsystem to process based on the called method.
D1	The Display subsystem sends a visual representation of the current GUI screen to the touch screen to be seen by the customer.
D2	The Display subsystem sends a visual representation of the current GUI screen to the touch screen to be seen by the administrator.
OA1	The Object Avoidance subsystem sends a structure of data indicating the direction of the object that needs to be avoided and the distance to it to the Flight Dynamics subsystem.
FD1	The Flight Dynamics subsystem sends a structure of data indicating the power level for each motor to Motor Control

DC1	The Drone Control subsystem sends a structure of data indicating how the drone should move to the Flight Dynamics subsystem.
DC2	The Drone Control subsystem sends an integer value representing a color and sound to the Light/Sound subsystem.
DC3	The Drone Control subsystem sends a status message back to the User Interaction subsystem.
UI1	The User Interaction subsystem sends data for queries to the Database Manager subsystem to pull or store from the database based on the method call.
UI2	The User Interaction subsystem sends data to show in the GUI to the Display subsystem.
UI3	The User Interaction subsystem sends a series of X Y coordinate waypoints that the drone would need to follow to reach the desired position to the Drone Control subsystem.
AI1	The Admin Interaction subsystem sends a new point to navigate to, with X and Y values, to the Path Finding subsystem.
AI2	The Admin Interaction subsystem sends data to show in the GUI to the Display subsystem.
AI3	The Admin Interaction subsystem sends data for queries to the Database Manager subsystem to pull or store from the database based on the method call.
PF1	The Path Finding subsystem sends a series of X Y coordinate waypoints that the drone would need to follow to reach the desired position to the Admin Interaction subsystem.
DM1	The Database Manager subsystem sends objects containing the information requested to the Admin Interaction subsystem.
DM2	The Database Manager subsystem sends a formatted query to the physical Data Storage.
DM3	The Database Manager subsystem sends objects containing the information requested to the User Interaction subsystem.
DS1	The physical Database Storage returns data from the request to the Database Manager subsystem.

2.2.3 Client Presentation

The Client Presentation Layer controls the appearance and execution of code on the user kiosk. It accepts user input through the touchpad and updates the display as necessary. The Client Presentation Layer is responsible for search functionality, standard output for all users, and administration interfaces. The graphical user interface will mediate all functionality of this layer, acting as the public interface of the system. It sends any processing tasks to the next layer down, the Client Processing Layer.

2.2.4 Client Processing

The Client Processing Layer handles all requests from the Client Presentation Layer. It has specific calls that can be made to set off processes for most of the use cases of the system. These include searching for items, adding items to the store, and adding shelves to the layout. In addition it will handle the initial set up of the store. It needs to talk to the Data Storage Layer for information on the items and store and has to send requests to Drone Processing to start drone navigation. It also listens for messages from Drone Processing such as when the drone completes a search or runs out of power.

2.2.5 Data Storage

The Data Storage Layer maintains the information on stores, item locations, and paths necessary for efficient system execution. Information on item locations captures data such as location, price, category, and stock. Information on the store includes exact layout. This layer stores a path to each item as a series of waypoints. All information functions in this layer are controlled by a database manager, responsible for translating appropriate requests into specific queries.

2.2.6 Sensor Input

The Sensor Input Layer is responsible for monitoring and communicating the status of system sensors. Some sensors are mounted on the drone, and others are placed around the store. One set of store sensors use Infrared to detect the drone's position based on its onboard Infrared emitter. Another set of onboard sensors use ultrasonic to monitor proximity while the drone is in flight. One sensor on the drone and one sensor on the base station coordinate to line up landings after a mission. Another drone sensor monitors its power level. All of these sensors will have standardized input based on manufacturer specifications. This layer will mediate the standard inputs and send appropriate status reports to Drone Processing.

2.2.7 Drone Processing

The Drone Processing Layer controls the navigation and control of the drone as it navigates the store. It uses input from the Sensor Input Layer to monitor the drone's location and activate obstacle avoidance if necessary. It uses instructions from the Client Processing Layer about where to go and sends back simple data about the status of the drone. Navigation commands call to the Drone Hardware Output Layer to tell it to activate the drone's hardware components.

2.2.8 Drone Hardware Output

The Drone Hardware Output Layer is responsible for the physical systems that keep the drone active. It drives the individual motors on the drone and activates lights and sounds. The Drone Hardware Output Layer accepts instructions from the Drone Processing Layer. Its outputs are physical responses such as flashing lights or a turn during flight.

2.3 Detailed Design Specification

2.3.1 Module Decomposition Diagram

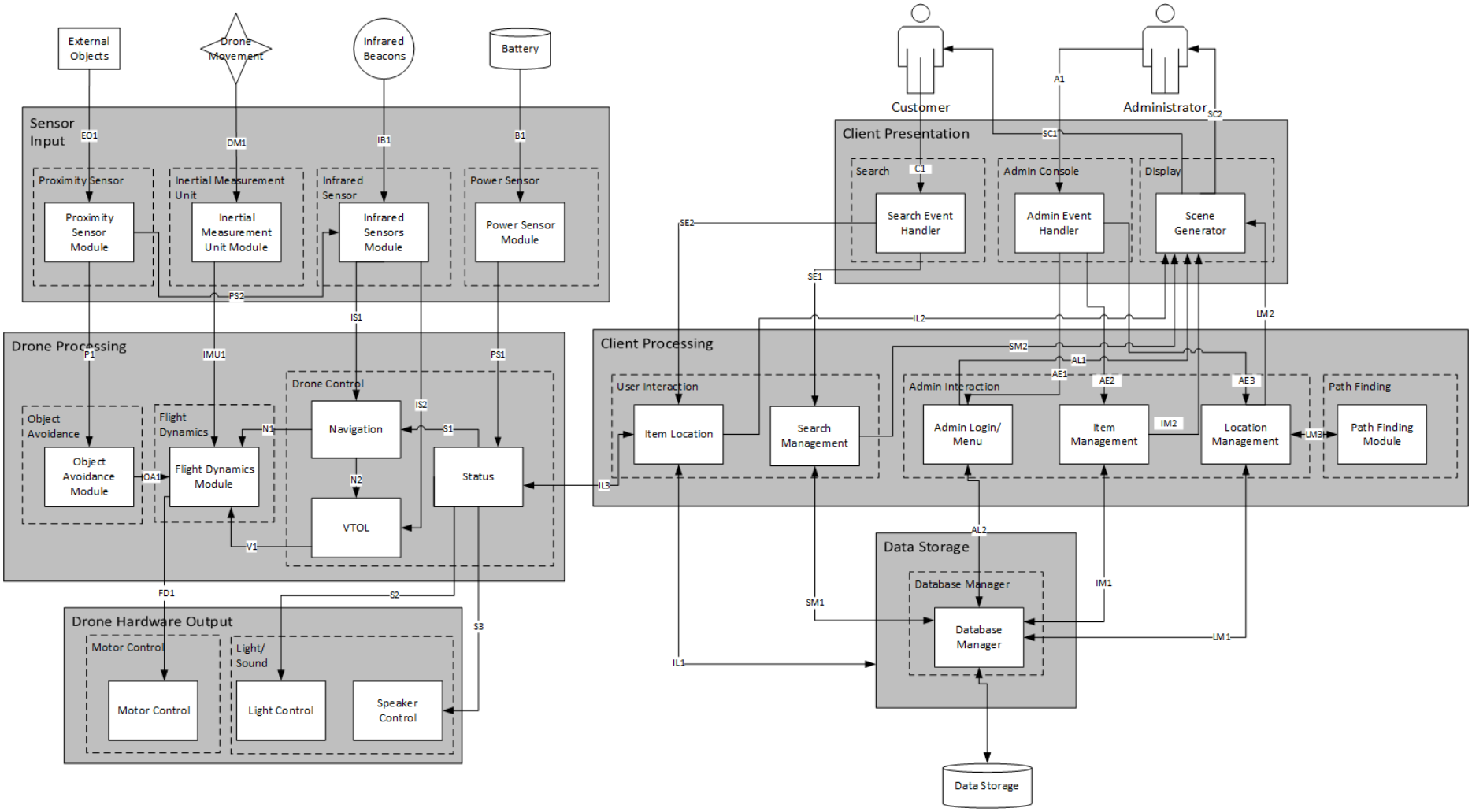

Figure 2-2 Module Decomposition Chart

2.3.2 Producer Consumer Matrix

Producers \ Consumers	External Objects	Drone Movement	Infrared Beacons	Battery	Customer	Admin	Proximity Sensor	Inertial Measuerment Unit	Infrared Sensors	Power Sensor	Search Event Handler	Admin Event Handler	Scene Generator	Object Avoidance	Flight Dynamics	Navigation	VTOL	Status	Item Location	Search Management	Admin Login/Menu	Item Management	Location Management	Path Finding	Motor Control	Light Control	Speaker Control	Database Manager
External Objects							EQ1																					
Drone Movement								DM1																				
Infrared Beacons									IB1																			
Battery										B1																		
Customer											C1																	
Admin												A1																
Proximity Sensor									P2					P1														
Inertial Measuerment Unit															IMU1													
Infrared Sensors																IS1	IS2											
Power Sensor																		PS1										
Search Event Handler																			SE2	SE1								
Admin Event Handler																					AE1	AE2	AE3					
Scene Generator					C2	A2																						
Object Avoidance															OA1													
Flight Dynamics																									FD1			
Navigation															N1		N2											
VTOL															V1													
Status															S1											S2	S3	
Item Location													IL2					IL3										IL1
Search Management													SM2															SM1
Admin Login/Menu													AL1															AL2
Item Management													IM2															IM1
Location Management													LM3											LM3				LM1
Path Finding																												
Motor Control																												
Light Control																												
Speaker Control																												
Database Manager																												

Figure 2-3 Producer-Consumer Matrix

2.4.1 Requirements Traceability

Table 2-8 Requirements Traceability Matrix

Req. #	Requirement Name	Search Event Handler	Admin Event Handler	Scene Generator	Proximity Sensor	IMU	Infrared Sensors	Power Sensor	Search Management	Item Location	Admin Login/Menu	Item Management	Location Management	Path Finding	Status	Navigation	VTOL	Object Avoidance	Flight Dynamics	Database Manager	Servo Controller	L.E.D. Controller	Speaker Controller
3.1	Search By Name	x		x					x	x					x					x			
3.3	Search By Category	x		x					x	x					x					x			
3.4	Follow Drone	x		x	x	x	x	x		x					x	x	x	x	x		x	x	x
3.6	Open Administrator Console		x	x							x												
3.7	Manage Items in Inventory		x	x								x								x			
3.10	Make Drone Visible/Audible														x							x	x
3.12	Service Drone																x						
5.1	Drone Response Time														x						x		
5.3	Drone Travel Time				x	x	x	x								x	x	x	x		x		
5.4	Return After Completion of Guidance				x	x	x	x								x		x	x		x		
5.5	Base Landing Upon Completion																x		x		x		
8.1	Configuration of Store		x	x									x	x						x			
8.2	Date Storage											x	x							x			

3. Test Items

This section covers the steps necessary to test the product from unit to system levels. Testing begins with individual hardware components for the drone. Next, the team unit tests each module within the system on both client and drone sides. These aggregate into the component tests at the module level. Modules combine during integration testing at the layer level. Finally, the team completes overall system validation.

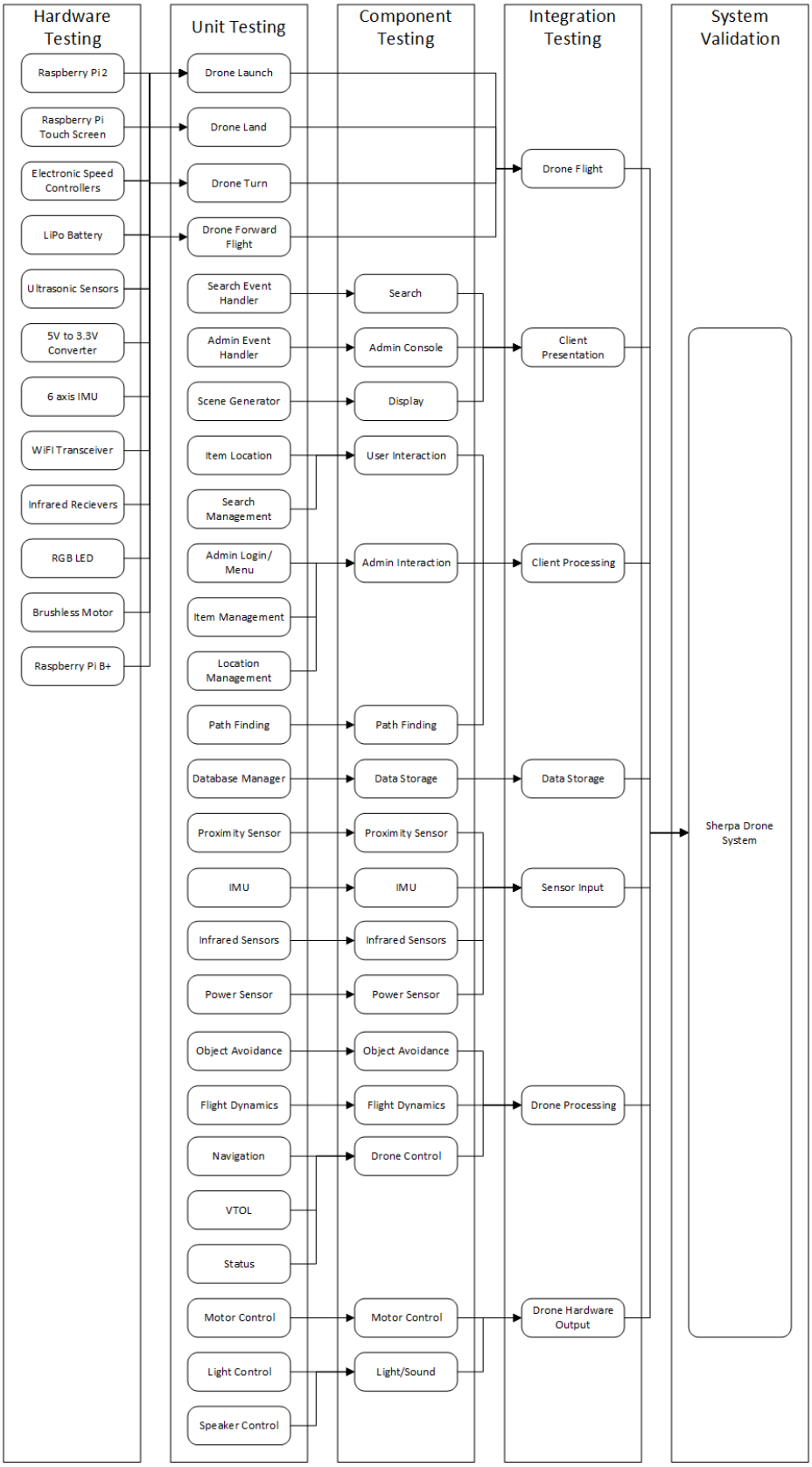

Figure 3-1 Testing Composition Diagram

3.1 Hardware Tests

Table 3-1 Hardware Tests

Test ID	Hardware	Input	Output	Test	Priority
H1	Raspberry Pi 2	Power	System boots and runs startup software	Apply power to the system and observe response through monitor interface	Critical
H2	Raspberry Pi LCD Touch Screen Display	Power	System boots and awaits user interaction	Apply power to the system and observe response through monitor interface	Critical
H3	Electronic Speed Controllers	Electric signal	Speed command	Input speed signal and verify that controller modulates.	Critical
H4	3S LiPo Battery	Power charge	Battery power	Charge battery to full and test voltage and duration of power	High
H5	Ultrasonic Sensors	Object at some distance	Distance to the object in inches	Place objects a known distance from the sensor and corroborate the reading with distance	High
H6	5V to 3.3V I2C Converter	5 volts	3.3 volts	Test voltage levels on either side of converter while active	Medium

H7	6 axis IMU	Change to physical orientation of sensor or its speed	Array of float accelerations in X, Y, Z and the gyroscope reading	Adjust X, Y, Z orientation of the IMU at differing accelerations and observe output.	Critical
H8	WiFi Transceiver	Signal to transmit over WiFi frequency	Radio frequency signal	Transmit a predetermined signal across the transceiver and verify integrity at the other end	Medium
H9	Infrared Emitter	Positional signal	Distance from emitter to infrared receiver	Place emitter and receiver at known distances and verify the output of the emitter	Medium
H10	RGB LED	Signal directing which lights to activate	Light activation	Send signal for light activation and verify that such occurs	Low
H11	2212 930KV Brushless Motor	Input speed command	Rotation at speed	Input signals for various speeds and verify motor activates as commanded	Critical
H12	Raspberry Pi Model B+	Power	System boots and runs startup software	Apply power to the system and observe response through monitor interface	Critical

3.2 Unit Tests

Table 3-2 Unit Tests

Test ID	Module	Input	Output	Test	Priority
U1	Search Event Handler	User touch event	User input as Java data	Use kiosk touchscreen to create touch events and verify the Java output	Critical
U2	Admin Event Handler	Admin touch event	Admin input as Java data	Use kiosk touchscreen while logged in as admin to create touch events and verify the Java output	Critical
U3	Scene Generator	Data from Client Processing modules	JavaFX Scene output to kiosk screen	Feed predetermined search result data to module and verify the correct Scene result	Critical
U4	Item Location	ID of Item	Path object with Waypoints for Drone Status	Compose a valid Item ID and verify integrity of Path object result	Critical
U5	Search Management	Search string	Command to Database manager	Compose a search string and verify the correct command for the database manager	Critical

U6	Admin Login/Menu	String password	Command to Scene Generator to show admin menu	Input the hard coded admin password and verify the administrator menu appears	Critical
U7	Item Management	Item ID and method call to insert, update, delete	Update to item or status to display on scene generator	Create an item, update the item, and delete the item. Check that each step has the appropriate status report.	Critical
U8	Location Management	Store layout information	Path and Location to save in the database	Modify the store layout with an inaccessible square. Check that path and location objects correspond.	High
U9	Path Finding	Location object, kiosk location, and store layout	Path object to Location object	Send a Location object to module and verify the Path object returns a shortest path.	Critical
U10	Database Manager	Query command	Status result and optional object	Input database commands through bridge interface. Verify through independent SQL commands that the changes are reflected in the storage.	High

U11	Proximity Sensor	External objects	Float representing distance to any object in inches	Manually place object in front of proximity sensor and verify reported distance	High
U12	Inertial Measurement Unit	Change to physical orientation of sensor or its speed	Array of float accelerations in X, Y, Z and the gyroscope reading	Adjust X, Y, Z orientation of the IMU at differing accelerations and observe output.	Critical
U13	Infrared Sensors	Infrared data	Distance from emitter to infrared receiver	Place emitter and receiver at known distances and verify the output of the emitter	High
U14	Power Sensor	Power charge	Battery power	Charge battery to full and test voltage and duration of power	Medium
U15	Object Avoidance	Proximity sensor distance data	Override command for Flight Dynamics	Verify that object avoidance sends the override command to Flight Dynamics when an object is placed closer than the critical threshold before the drone.	Critical
U16	Flight Dynamics	IMU and positional vectors	Motor speed vectors	Supply the module with virtual IMU and positional vectors. Verify that the motor speed vectors reflect appropriate flight.	Critical

U17	Navigation	Current waypoint from Status and current position from Infrared sensors	Speed and direction to travel	Supply a waypoint and position to module. Verify that speed and direction are consistent with safe travel to that waypoint.	Critical
U18	VTOL	Infrared sensor landing distance	Direction and speed to travel	Supply the module with distance to landing and verify its directions for landing.	High
U19	Status	Waypoints from kiosk and power level of battery	Current waypoint to travel to	Supply the module with waypoint and power level data. Verify that waypoints are sent in appropriate order and integrity	Critical
U20	Motor Control	Float array of Cartesian direction and speed	Pulse-width Modulation signal to motor	Given the Cartesian direction and speed, test that the PWM signal remains within safety tolerance and coordinates action between the four motors	Critical
U21	Light Control	Character representation	Physical light activation	Send the module a character and visually verify the activation of the hardware	Medium

U22	Speaker Control	Byte to cause activation	Physical activation of the speaker	Verify that a byte of zero does not activate and that a byte of one activates the speaker. Verify that volume is within acceptable limits.	Medium
U23	Drone Launch	Command to launch from base station	Drone movement	Drone lifts off from base station and maintains stable flight	Critical
U24	Drone Forward Flight	Command to move one square unit forward	Drone movement	Drone moves a square unit forward in a straight line at stable altitude	Critical
U25	Drone Turn	Command to turn ninety degrees left or right	Drone movement	Drone turns to left or right by ninety degrees without loss of altitude or lateral movement	Critical
U26	Drone Land	Command to land on base station	Drone movement	Drone execute landing on base station and comes to a complete stop	High

3.3 Component Tests

Table 3-3 Component Tests

Test ID	Subsystem	Input	Output	Test	Priority
C1	Search	User action	Search request	Input a search through the touchscreen and verify that sends correct request to search management	Critical
C2	Admin Console	Admin action	Store update	Log in as admin and request update of item. Verify that request is made of item management.	Critical
C3	Display	Data from Client Processing modules	JavaFX Scene output to kiosk screen	Feed predetermined search result data to module and verify the correct Scene result	Critical
C4	User Interaction	User request string	Request result	Make a user request through the touchscreen. Verify that request reaches database, and that database returns the correct tuples.	Critical

C5	Admin Interaction	Admin request string	Request result	Make an admin request. Verify that request reaches database, and that database updates state.	Critical
C6	Path Finding	Layout file and location	Path sequence from kiosk to location	Request a Path item for a location and layout. Verify that the series of Waypoints is viable.	High
C7	Data Storage	Database state request	Set of tuples or update flag	Accept a search request and verify that tuples comply. Accept an update request and verify that state updates.	High
C8	Proximity Sensor	Object distance	Object distance	Verify that object distance is transmitted to object avoidance module without error.	High
C9	Inertial Measurement Unit	Six axis orientation	Six axis orientation	Verify that orientation is passed to Flight Dynamics without error	Critical
C10	Infrared Sensors	Relative position	Relative position	Verify that drone position information is passed to the Navigation and integrated	Critical

C11	Power Sensor	Current power level	Current power level	Verify that drone battery information is relayed to Status without error	High
C12	Object Avoidance	Object proximity	Flight Dynamics override	Given an object in near proximity, verify that Object Avoidance fires the override to assume control of drone flight	High
C13	Flight Dynamics	Six axis orientation, object avoidance, navigation and VTOL	Motor flight vectors	Verify that module correctly shifts flight procedure by signals from object avoidance, navigation, and VTOL	Critical
C14	Drone Control	Infrared position, waypoints, power level	Waypoint command to Flight Dynamics	Verify that Infrared, waypoint, and power data meld to form correct command to Flight Dynamics	Critical
C15	Motor Control	Motor control vectors	Motor movement	Verify that motor vectors translate into correct rotation and speed of each motor	Critical
C16	Light/Sound	Light and sound command	Light and sound activation	Verify that Status signal translates into correct colors and sound activation	High

3.4 Integration Tests

Table 3-4 Integration Tests

Test ID	Layer	Input	Output	Test	Priority
I1	Client Presentation	User or administrator touch events	JavaFX Scene appropriate to request	Pass user command to search for a specified object. Verify that system searches for item and reports status. Command the system to launch the drone, and verify that the command is sent.	Critical
I2	Client Processing	Requests from user or administrator to access data or request drone	Items, Paths, and status codes for user and drone	As administrator, command the system to add a new location. Add a new item at this location. Log out of administrator. As user, search for that item and send the drone.	Critical

I3	Data Storage	Requests from Client Processing for data	Query data with status updates	As administrator, command the system to add a new location. Add a new item at this location. Through independent SQL, verify that the location, its path, and its object all exist in the database.	High
I4	Sensor Input	Physical environment	Distance, orientation, and power information	Manipulate the distance of obstructions around the drone between one inch and ten feet. Verify that drone orientation is correct during a banked turn. Verify that drone recognizes a low battery condition.	Critical
I5	Drone Processing	Distance, orientation, power, and waypoint information	Motor vectors and light/sound commands	Command the drone to fly to an item location. Verify that the drone makes correct navigational choices to arrive at specified location.	Critical

I6	Drone Hardware Output	Motor vectors and light/sound commands	Movement of drone with light and sound activation	Verify that drone maintains altitude, orientation, and speed during flight.	Critical
I7	Drone Flight	Control statements	Physical drone movement	The drone moves when commands are issued from a console	Critical

3.5 System Validation

Table 3-5 System Validation

Test ID	Validation	Input	Output	Test	Priority
S1	Customer can search for an item	Input to the kiosk touchscreen and search of screens	Sequence of screens to find the item. Drone will launch and navigate store.	Use the kiosk interface to find an item of interest. Select the item. Verify that drone launches within thirty seconds and navigates at walking pace to the object. It must activate light and sound at the location.	Critical

S2	Administrator can add an item to the store	Input to the kiosk touchscreen to log in and add item	Sequence of screens to add an item.	Use the kiosk interface to select the add item option and enter details. Exit administrator mode. Verify that item exists for customer use.	Critical
S3	Administrator can update an item in the store	Input to the kiosk touchscreen to log in and update item	Sequence of screens to update the item.	Use the kiosk interface to select the update item option and enter details. Exit administrator mode. Verify that item has updated for customer use.	High
S4	Administrator can initially set up the store	Input to the kiosk touchscreen to log in and set up store	Sequence of screens to set up the initial store.	Use the kiosk interface to select the set up store option and enter details. Select the add item option and verify that the store has locations such that items may now be added.	Medium
S5	Administrator can add a location to the store	Input to the kiosk touchscreen to log in and add location	Sequence of screens to adjust store layout.	Use the kiosk interface to select the update store option and enter details. Select the add item option and verify that the store locations have changed as desired.	High

S6	Follow Drone	A search command from the kiosk	Drone will launch and navigate to the item	Initiate drone guidance. Follow the drone at a walking pace. Verify that the drone navigates the store such that the user has full sight of the drone and does not need to make any detours to follow.	Critical
S7	Drone Return	Completion of guidance	Drone returns on its own to the kiosk and lands	Initiate drone guidance. At completion, verify that drone returns with no user prompting to the kiosk and successfully lands. Verify that the drone is now available for new user commands.	Critical

4. Risks

This section discusses the potential risks that the team may encounter during the testing phase of the Sherpa Drone. The risks are split between three categories: High, Medium, and Low. Controlling for risk is a major priority for this phase since a setback can cost the team weeks of work.

4.1 Risks Table

Table 4-1 Risks Table

Risk ID	Risk	Impact	Severity	Mitigation Strategy
R1	Motors burn out	The drone's motors burn out during testing	Low	Buy multiple spares of the motors
R2	Raspberry Pi is unable to run software	Raspberry Pi cannot run the software fast enough	High	Make the program multithreaded to fully utilize the Pi
R3	Wiring comes lose	The wiring comes lose during testing	Low	Ensure the wiring is secure
R4	Propellers break	The propellers break during the testing	Low	Buy multiple spares of the propellers
R5	Sensor Nodes fail to communicate	The signal from the nodes fails to reach the base station	High	Come up with an alternate way for the node to communicate
R6	Base Station fails to communicate with Drone	The drone fails to receive any information from the base station	Medium	The drone goes to the designated safe zone.

5. Features To Be Tested

This section details the requirements and features that will be implemented in the final project and the Testing criteria that will be used to ensure full usability.

5.1 Customer Requirements

5.1.1 Search By Name

Description: The system shall allow a customer to type in the name of an item to search for at the kiosk.

Testing Approach: The team will turn on the system and input names of items located in the database. The team will check the results against the actual contents of the database and test if it will return all the proper results

5.1.2 Search By Category

Description: The system shall allow a customer to select an item from the search list based on category at the kiosk. This will provide them with a list of all items in the store in that category that they can browse to select.

Testing Approach: The team will turn on the system and input a category into the search bar. The team will test all the categories available in the system.

5.1.3 Follow Drone

Description: The system shall allow a customer to follow a drone around the store towards the desired item location. The drone must fly at a reasonable pace and travel a path that the customer can follow without difficulty.

Testing Approach: The team will input the name of an item into the kiosk and send the drone out to find the item. The team will then follow the drone to ensure that it is flying at an appropriate speed.

5.1.4 Open Administrator Console

Description: The system shall allow an administrator to log in, with a password, to a special menu at the kiosk in order to add, remove, and move items in the store.

Testing Approach: The team will turn on the system and log into the administrator console. If there is no response, the test will be considered a failure.

5.1.5 Manage Items in Inventory

Description: The system shall allow an administrator to add and modify items to the store with their name, category, and location at the kiosk. This will include adding a new category if none exists that works for this item.

Testing Approach: The team will turn on the system and log into the administrator console. Then the team will try to add items, remove items, and manipulate items by name, category, and location. The team will then verify that the system has saved those changes.

5.1.6 Make Drone Visible/Audible

Description: The drone will use an LED light and sound cues to announce its activity to surrounding customers. The LED will support multiple colors so that if the drone is helping a customer and another approaches the kiosk and searches for an item it will inform them that, when the drone will lead them, it will have a different color.

Testing Approach: The team will turn on the system, type in an item, and have the drone go to the item. The team will then see if the drone's LED lights and speakers are flashing and beeping respectively.

5.1.7 Service Drone

Description: The system shall allow an administrator or service personnel to perform maintenance on the drone, including charging the battery.

Testing Approach: The team will change the battery on the drone and ensure parts are available to modify.

5.2 Packaging Requirements

5.2.1 Kiosk Packaging

Description: The kiosk will be enclosed in a soft edged metal box with a 7 inch touch screen on the front, a flat landing zone on top, and a power connection in the back.

Testing Approach: The team will ensure that that kiosk is enclosed by building the enclosure, placing the touch screen in it, then plugging the system in. Note further design has ruled that a metal box is a poor choice and the final construction will be a different material.

5.2.2 Drone Packaging

Description: The Drone will be a quad-copter style drone consisting of four rotors, a small frame, an RGB LED, a small speaker, a Styrofoam cover, and a central control board.

Testing Approach: The team will ensure that the drone will be a quad-copter by building it as such, with a small frame, and RGB LED, a small speaker, a Styrofoam cover, and a central control board.

5.2.3 Drone Sensors

Description: The drone will include ultrasonic sensors for the detection of objects in its vicinity.

Testing Approach: The team will test the ultrasonic sensors by placing objects in front of the sensor and ensuring that the sensors detect the objects and the distance from those objects.

5.2.4 System Installation

Description: The system designers will install the necessary product pieces at the retail environment.

Testing Approach: The team will test the system installation by installing the system ourselves and ensuring that the kiosk and drone are as modular as possible.

5.3 Performance Requirements

5.3.1 Drone Response Time

Description: The drone will demonstrate responsive behavior through a light that turns on to indicate a message received. Messages from the base station to the drone will be delivered in an interval of under a second to allow prompt drone response.

Testing Approach: The team will turn on the system and verify that the connection from the base station to the kiosk is working properly. The team will then send the drone messages and measure the response time of the drone reacting to those messages.

5.3.2 Drone Wake Time

Description: The system will respond to a new item selection and wake the drone from a dormant state on the base station within fifteen seconds. Total time to calculate route, communicate with the drone, and begin travel must fall within this time.

Testing Approach: The team will turn on the system and send the signal to the drone to look up an item. Then the team will measure the time it takes the drone to go from a dormant state to preparing to take off.

5.3.3 Drone Travel Time

Description: The drone will complete its travel in a reasonable time span as appropriate to the size of the store and the maximum number of items allowed in a single trip.

Testing Approach: The team will turn on the system and send the signal to the drone to look up the shortest path to an item, then to fly to it. The team will measure the time it takes for the drone to go to the item and return to the base station.

5.3.4 Return After Completion of Guidance

Description: The drone will return to the base station once travel has been executed for the customer. The drone will not be expected to wait on customer input. It may follow a path that is not appropriate to customers.

Testing Approach: The team will turn on the system and send the signal to the drone to look up an item, then to fly to it. The team will measure the time it takes for the drone to go from the item to the base station.

5.3.5 Base Landing Upon Completion

Description: The drone will execute a landing on the base station when guidance is complete. If the drone loses connectivity with the base station it will land in a designated safe zone in the store.

Testing Approach: The team will turn on the system and send the signal to the drone to look up an item, then to fly to it. When the drone returns, the team will ensure that the drone properly lands on the base station without falling or crashing.

5.3.6 Minimum Operational Time

Description: The drone will be able to maintain operational flight time of at least thirty minutes from a fully charged state.

Testing Approach: The team will have the drone run a continuous series of item look ups after ensuring that it is fully charged, then will measure the time it for the battery to be unable to operate. Note, the team does not expect in this prototype to meet this requirement and is optimistically hoping for 15 minutes of flight time.

5.3.7 Single Drone Supported

Description: The system will support the full operation of a single drone at a time. This drone will be responsible for guiding all customers.

Testing Approach: The team will build only a single drone and ensure that the signal from the drone to the base station does not interfere with any other signal that could be used by other drones to communicate with other base stations.

5.3.8 Operational Range

Description: The drone will maintain operational contact with the base station at all times through a wireless signal. Any drone that loses contact with the base station through exceeding the operational range or interference will immediately land and power down in the designated safe zone.

Testing Approach: The team will send the drone to look up an item, then cut the signal from the base station. The team will then observe if the drone follows its programming and returns to the base station.

5.4 Safety Requirements

5.4.1 Drone Rotors

Description: The drone rotors shall not harm a person or damage property.

Testing Approach: The team will test this function by having the rotors spin at takeoff speed and measure the force they generate.

5.4.2 Drone Weight

Description: The system shall be light enough to avoid damage to person or property if it were to fail.

Testing Approach: The team will measure the weight of the drone.

5.4.3 Property Damage

Description: The system shall be designed so that it does not cause property damage. A collision with another object does not cause property damage.

Testing Approach: The team will ensure that force generated by drone in flight or while following is low enough to not hurt objects or people.

5.4.4 Eye and Ear Protection

Description: The system shall not cause damage to peoples' eyes through its LED or to peoples' ears through the speaker.

Testing Approach: The team will test the LED and ensure that its maximum brightness does not exceed the pain threshold. The team will test the speaker and ensure that its maximum volume does not exceed the pain threshold.

5.4.5 Drone Avoids Collisions

Description: The drone shall fly in such a manner to avoid pedestrians or objects.

Testing Approach: The team will send the drone to look up and item and have obstacles preplaced. The team will see if the drone follows its programming and avoids the obstacles to go to its objective

5.4.6 Altitude Control

Description: The system shall control its flight altitude to avoid hazards to customers or the environment. The flight altitude shall be set in the initial configuration of the product for a store. At this time the altitude shall be limited to a single level, i.e. a store with two floors would require separate drones for each floor.

Testing Approach: The team will ensure that the drone stays on a single level by sending it to look up an item and measuring any variance in height.

5.4.7 Down Draft

Description: The system shall be designed so that rotor down draft does not cause damage or harm to person or property.

Testing Approach: The team will set the drone to hover and measure the force generated by the rotors.

5.5 Maintenance and Support Requirements

5.5.1 Source Code and Documentation Availability

Description: All source code developed by Ground Control will be made available to the maintenance team and future development teams. This source code will include adequate comments to document its functionality. The team will make available all schematics and technical documentation, including: System Requirements Specification, Architectural Design Specifications, Detailed Design Specifications, and System Test Plan.

Testing Approach: The team will ensure that all source code is properly documented and will make sure that all the documentation is available online.

5.5.2 User Manual

Description: The User Manual will contain instructions for operation of the system. It will instruct administrators in maintaining an accurate layout of their store and inventory of the items in the store. Another section will instruct customers on how to use the kiosk, which will be accessible from the kiosk itself.

Testing Approach: The team will write a user manual for the drone, then have people unfamiliar with the project set up the kiosk and the drone.

5.6 Other Requirements

5.6.1 Configuration of Store

Description: The system will allow the installation team to input the layout of the store into its memory. This will include where the aisles and other locations (tables, racks, etc.) where items are located in the store.

Testing Approach: The system be turned on and the team will open the administer console. Then the team will then input the layout of the store and will ensure that the system has verified that it has been updated.

5.6.2 Data Storage

Description: The system will store inventory information in a database. This database will include the location data necessary for drone path-finding. It will also include category of each item to assist in user searches.

Testing Approach: The system be turned on and the team will open the administer console. The team will then input the item into the database by filling out the proper information. Then the team will insure that the system has verified that it has been updated.

6. Features Not To Be Tested

This section describes the features that will not be included in the final product and the reasoning behind the decision.

6.1 Customer Requirements

6.1.1 Auto-Fill Search Bar

Description: The system shall suggest possible items to the customer as the customer types into the search bar at the kiosk.

Reasoning: Concerns about processing power on prototype machine and development time constraints.

6.1.2 Stay with Customer

Description: The drone will match pace with the customer, slowing as necessary. If the customer falls behind, the drone must stop to wait. The drone must wait for the customer until the customer approaches again or thirty seconds have passed when the customer will be informed to start moving and if an additional fifteen seconds pass the system will cancel the search.

Reasoning: Future requirement never expected to be tested.

6.1.3 Track Inventory Changes

Description: The system shall allow an administrator to update the locations of items automatically using RFID or other inventory tracking technology.

Reasoning: Future requirement never expected to be tested.

6.1.4 Mobile App Access

Description: The system shall include a mobile app that allows customers to request the service of a drone while within the store.

Reasoning: Future requirement never expected to be tested.

6.2 Packaging Requirements

6.2.1 Kiosk circuitry

Description: Kiosk enclosure will house all required circuitry, wires and components out of sight from the customer.

Reasoning: Prototype is not expected to fully meet this requirement.

6.2.2 Kiosk Power Source

Description: Kiosk will utilize a single external power cable. This cable will provide the correct DC power to components from an AC plug.

Reasoning: Future requirement never expected to be tested.

6.2.3 Drone Circuitry

Description: Drone circuitry, including the control board, will be housed entirely out of sight of the customer.

Reasoning: Prototype is not expected to fully meet this requirement.

6.2.4 Drone Wires

Description: Drone will enclose most wires from customer vision. The only acceptable exposed wires are those to the motors and from the batteries.

Reasoning: Prototype is not expected to fully meet this requirement.

6.3 Performance Requirements

6.3.1 Max Items Per Trip

Description: The system will require that users specify no more than an allotted number of items per guidance operation. This number will be set by the administrator as part of system set up. Customers who wish to follow the drone to more items must return to the base station and complete another request. This limit will prevent a single customer request from monopolizing the use of the drone.

Reasoning: Time constrains do not allow testing of this feature.

6.3.2 Base Station Charging

Description: The drone will execute a landing on the base station in the correct orientation to charge. If unable to charge, it will attempt a second landing. If this landing fails, the drone will land on the base station and notify the administrator.

Reasoning: Time constrains do not allow testing of this feature.

6.4 Safety Requirements

6.4.1 Damage Avoidance

Description: The system shall be designed to fail safely. Damaged parts shall not pose an immediate hazard to person or property.

Reasoning: Time and cost constrains do not allow testing of this feature.

6.5 Maintenance and Support Requirements

6.5.1 Drone Maintenance

Description: Standard maintenance of the drone, such as checking for damaged parts, will be included in the User Manual. The Ground Control team will train technicians to take over maintaining the system after completion of the project. If a software update is needed for the drone, the trained technicians will complete installation.

Reasoning: No team of technicians will take over for the project.

6.5.2 Kiosk Maintenance

Description: Standard maintenance of kiosk will be included in User Manual. The Ground Control team will train technicians to take over maintaining the system after completion of the project. For software updates, administrators will be able to download the update onto a flash drive and install it on the kiosk.

Reasoning: No team of technicians will take over for the project.

6.5.3 Trouble shooting Guide

Description: The troubleshooting guide will include solutions to commonly detected problems discovered during system testing and from users of the Sherpa Drone.

Reasoning: No team of technicians will take over for the project.

6.6 Other Requirements

6.6.1 Protected Administrator Console

Description: The system will hide the administrator console from access by the customer by using a special code or sequence to bring up the log in page.

Reasoning: Future requirement never expected to be tested.

7. Testing Approaches

This section details the methodologies the team will use to test the Sherpa Drone. This includes the strategies and tools the team will use for testing as well as the core functionality that must function to deliver the system and the metrics the team will use to determine if a test has passed.

7.1 Strategy

The testing phase of the system will take place both during and immediately after development. unit tests will be performed while writing code or shortly after it is complete and will be used to iron out immediately apparent issues with the code. Towards the end of development additional testing will be performed to catch integration bugs or other subtle bugs not found during development.

Hardware items will be tested as they arrive from our suppliers. Some components will have to be combined in order to test that they function as expected. This process will be done as soon as the parts become available to catch hardware that does not perform as expected so that other hardware can be ordered as replacements.

7.2 Tools

The team will use a variety of tools for testing the system including:

- JUnit
- SQLUnit
- Excel spreadsheets (Manual integration test cases)
- SciPy simulations

7.3 Core Functionality

The core functionality of the Sherpa Drone system is as follows:

- Search for item by name
- Search for item by category
- Follow drone to item
- Add item to store
- Update item in store
- Set up store with size and locations

7.4 Test Metrics

The metrics used to determine if a requirement has passed vary based on the priority level of the requirement as seen below.

Table 7-1 Metrics by Priority

Priority	Description	Success Criteria	Failure Criteria
Critical	Functions that are necessary for the system to be considered complete.	100%	<100%
High	Functions that are expected functions of the system that provide basic improvements to functionality.	>=90%	<90%
Medium	Functions that are nice additions to the system but not necessary for functionality.	>=75%	<75%
Low	Functions that are not expected to be complete but would improve the system.	>=50%	<50%

8. Item Pass/Fail Criteria

This section lays out the criteria necessary to determine pass and failure on tests. It clearly defines the success or failure of each test.

8.1 Hardware Tests

Table 8-1 Hardware Tests

Test ID	Hardware	Pass Criteria	Failure Criteria
H1	Raspberry Pi 2	Turns on when power supplied. Runs Ground Control software on start up.	Fails to turn on when power supplied. Fails to start software.
H2	Raspberry Pi LCD Touch Screen Display	Turns on when power supplied. Detects and relays all touch events	Fails to turn on when power is supplied. Does not detect any touch event.
H3	Electronic Speed Controllers	Moderates speed	Does not moderate speed
H4	3S LiPo battery	Supplies sufficient power for drone flight	Does not provide sufficient power for full drone flight
H5	Ultrasonic sensors	Detect objects within range and relay signal	Does not detect objects within range or does not relay signal
H6	5V to 3.3V I2C converter	Transforms voltage from 5V to 3.3V	Does not translate to 3.3V

H7	6 axis IMU	Detects change to acceleration in X, Y, and Z directions	Fails to detect change in acceleration in any direction
H8	WiFi Transceiver	Transmits WiFi frequency signal Accepts WiFi frequency signal	Does not transmit WiFi frequency signal Does not receive WiFi frequency signal
H9	Infrared Emitter	Detects relative location to infrared receiver	Does not detect location of infrared receiver
H10	RGB LED	Flashes R, G, or B for specified duration on command	Fails to flash any light for the correct duration
H11	2212 930KV Brushless Motor	Maintains correct motor motion for entire flight	Fails to maintain correct motor motion
H12	Raspberry Pi Model B+	Turns on when power supplied. Runs Ground Control software on start up.	Fails to turn on when power supplied. Fails to start software.

8.2 Unit Tests

Table 8-2 Unit Tests

Test ID	Module	Pass Criteria	Failure Criteria
U1	Search Event Handler	Accepts user input to generate a search event Packages the search request for processing	Fails to accept user input Incorrectly packages request

U2	Admin Event Handler	Accepts admin input to generate a command Accepts admin log in Packages the command for processing	Fails to accept admin command Fails to transmit login Fails to package the command
U3	Scene Generator	Translate data into scene on the touch screen	Fail to generate correct data to scene
U4	Item Location	Translate request into a query to data storage Transmit waypoint data to drone status	Fail to query data storage for item information Fail to transmit correct waypoints to drone status
U5	Search Management	Translate search request into a query for data storage Return results of query to scene generator	Fail to query data storage for search Fail to transmit result to scene generator
U6	Admin Login/Menu	Verify given password Forbid authorization for incorrect password	Fail to verify correct password Allow access for incorrect password
U7	Item Management	Accept admin command for item changes and send to data storage Return results of changes to scene generator	Fail to accept command or send to data storage Fail to transmit result to scene generator
U8	Location Management	Accept admin command for location changes and send to data storage Request path object of path finding for new locations Return results of changes to scene generator	Fail to accept command or send to data storage Fail to request new path or supply information Fail to transmit result to scene generator

U9	Path Finding	Accept path request Return path object with waypoints for location	Fail to parse request Fail to return a path object with waypoints
U10	Database Manager	Accept requests for store state information Return objects with result Update state as commanded	Fail to accept requests for state information Fail to return correct result Fail to update state
U11	Proximity Sensor	Detect objects within range and relay signal	Does not detect objects within range or does not relay signal
U12	Inertial Measurement Unit	Detects change to acceleration in X, Y, and Z directions	Fails to detect change in acceleration in any direction
U13	Infrared Sensors	Detects relative position of drone in store	Detect incorrect or null position of drone in store
U14	Power Sensor	Take accurate reading of power remaining in battery	Fail to accurately monitor battery level
U15	Object Avoidance	Trigger object avoidance override when signal received for object	Fail to trigger object avoidance override
U16	Flight Dynamics	Receive command for movement and create flight vectors	Create incorrect or null vector for flight
U17	Navigation	Combine data from infrared sensors and waypoints from status Calculate discrete steps for drone flight	Fail to correctly combine data from sensors and status Fail to calculate correct discrete steps for flight

U18	VTOL	Send signal to flight dynamics module for control Supply flight dynamics with landing or take off data	Fail to take control of flight dynamics during landing Fail to supply correct landing or take off information
U19	Status	Accept waypoints from Client Processing Monitor signal from Power Sensor Notify navigation to begin flight Return status to client processing	Fail to receive waypoints from Client processing Fail to monitor power level and prohibit flight when low Fail to notify navigation to begin fly Fail to relay drone status in flight
U20	Motor Control	Accept flight vectors Activate all four motors as commanded	Fail to receive flight vectors Fail to activate any motor as specified
U21	Light Control	Translate signal into flashing lights	Fail to activate lights for duration
U22	Speaker Control	Translate signal into sound	Fail to activate sound Activate sound at incorrect decibel level
U23	Drone Launch	Drone launches from the base station and achieves stable flight at altitude	Drone fails to launch from base station Drone launches but does not maintain stable flight

U24	Drone Forward Flight	Drone moves a single store square forward in a straight line	Drone does not move forward Drone moves forward but zigzags Drone moves forward but alters altitude more than six inches
U25	Drone Turn	Drone turns left or right ninety degrees as ordered	Drone does not execute turn Drone executes turn at the wrong angle Drone executes turn but alters altitude Drone executes turn but drifts
U26	Drone Land	Drone lands on the base station and comes to a stable, complete stop	Drone fails to attempt a landing Drone lands on a surface other than the base station Drone lands but then shifts or falls from the base station

8.3 Component Tests

Table 8-3 Component Tests

Test ID	Subsystem	Pass Criteria	Failure Criteria
C1	Search	User string passed to search management	User string is altered or lost
C2	Admin Console	Admin string passed to admin menu	Admin string is altered or lost

C3	Display	Displayed result matches the database result or status update Scene activates with result	Displayed result is corrupted Scene activates with null or incorrect results
C4	User Interaction	User request transmits to database manager Database manager returns correct tuples	Request fails to reach database manager Database manager returns corrupted or null tuple
C5	Admin Interaction	Admin request transmits to database manager Database manager returns correct state	Request fails to reach database manager Database manager returns corrupted or null tuple
C6	Path Finding	Path item consists of waypoints that use open store space and customer can follow	Path item does not specify each step of path Path item returns null Path item traverses a blocked location
C7	Data Storage	Database manager implements changes to database state as requested Database manager returns status message or result tuple	Database manager fails to maintain state Manager fails to commit changes to state Database returns incorrect status or tuples
C8	Proximity Sensor	Transmits object distance measurement to Object Avoidance without error	Transmits incorrect or corrupted object distance measurement Fails to transmit data to Object Avoidance

C9	Inertial Measurement Unit	Transmits drone orientation measurement to Flight Dynamics without error	Transmits incorrect or corrupted orientation Fails to transmit data to Flight Dynamics
C10	Infrared Sensors	Transmits drone position measurements to Flight Dynamics without error	Transmits incorrect or corrupted position Fails to transmit position
C11	Power Sensor	Transmits accurate data on remaining battery life to Status	Transmits incorrect battery life to Status Fails to transmit battery life
C12	Object Avoidance	When given the object avoidance requirement, fires interrupt to the Flight Dynamics to assume control	Fires interrupt when object distance is greater than specification Fails to fire interrupt when an object is within distance
C13	Flight Dynamics	Integrates signals from object avoidance, navigation, VTOL	Fails to accept any signal from other modules Fails to accept interrupts from other modules
C14	Drone Control	Integrates signals from infrared sensors and power Interprets waypoint data to find next flight waypoint	Fails to integrate any signal Fails to provide a waypoint to Flight Dynamics Provides incorrect waypoint to Flight Dynamics
C15	Motor Control	Accepts motor vectors from Flight Dynamics	Fails to accept motor vectors

C16	Light/Sound	Accepts Status signal for activation	Fails to accept signal for activation Maintains activation beyond specification

8.4 Integration Tests

Table 8-4 Integration Tests

Test ID	Layer	Pass Criteria	Failure Criteria
I1	Client Presentation	Accept user input and create requests Display system updates to user Accept both user and admin interaction	Fail to read user input Fail to display system updates or scenes Fail to accept either user or administrator actions
I2	Client Processing	Translates client presentation requests into database requests Returns database results to presentation Calculates path to locations Sends waypoint data to drone	Fails to translate requests to database requests Fails to return database results to presentation Incorrectly calculates paths Fails to send correct waypoint data to drone
I3	Data Storage	Maintain store state in database format Respond to data requests with correct set of results	Corrupt or lose store state information Respond with incorrect or null results

I4	Sensor Input	Detect changes in objects around the drone Detect changes in drone location Accurately monitor power level	Fail to detect objects or hazards Fail to correctly detect drone location Fail to correctly monitor power level
I5	Drone Processing	Calculate flight data necessary for trip Calculate movement vectors Trigger object avoidance when object detected	Calculate incorrect flight coordinates Calculate incorrect movement vectors Fail to trigger object avoidance
I6	Drone Hardware Output	Output correct motor actions for flight Output light and sound signals	Fail to output correct actions for any of the motors Incorrect or null output for light and sound
I7	Drone Flight	Commands are obeyed with drone moving as expected	Drone does not accept commands Drone does not move as commanded

8.5 System Validation

Table 8-5 System Validation

Test ID	Validation	Pass Criteria	Failure Criteria
S1	Customer can search for an item	The user is able to search for an item	The user is not able to search for an item
S2	Administrator can add an item to the store	Administrator may add items to the store	The system fails to accept and maintain item information

S3	Administrator can update an item in the store	Administrator can update any item in the store	The system fails to accept and maintain item information
S4	Administrator can initially set up the store	Administrator can set up the store with an arbitrary layout	The system fails to accept and maintain location and path information
S5	Administrator can add a location to the store	Administrator can add a location to the store	The system fails to accept and maintain location information
S6	Follow Drone	The user is able to follow the drone through the store The drone maintains suitable speed	The drone takes a path that a customer cannot follow The drone travels too fast to follow
S7	Drone Return	The drone returns to the kiosk and lands after guiding a customer	The drone fails to return to the kiosk The drone fails to land on the kiosk

9. Test Deliverables

This section will detail the deliverables that the team will have upon completion of the testing process. The following subsections will describe what will be documented for each test case. All of these documents will be made available at the end of the project. The test cases, test case results, and defects log will exist as Excel spreadsheets within the source controlled folder for the team. When possible a member other than the developer will generate and run the test cases. Automated unit tests will be summarized as a single test case for each module, component, etc. under test.

9.1 System Test Plan

The System Test Plan will provide an overview of what will be tested, how these tests will be performed, and the pass/fail criteria dependent upon the expected results of the test.

9.2 Test Cases

Each test case will include the following:

- **Test Case ID:** Unique ID number for test case
- **Test Case Result ID:** Associate ID for the result of the test case
- **Author:** The team member that created the test case
- **Description:** a general description of the test case
- **Preconditions:** Any conditions the system needs to be in before running the test case
- **Post Conditions:** Any various condition the system can be in after running the test case
- **Test Steps:** Steps to run the test case, including any necessary inputs/outputs needed
- **Comments:** Any extra comments or special conditions needed to run test case

9.3 Test Case Results

Each test case result will include the following:

- **Test Case Result ID:** Unique ID number for test case result
- **Test Case ID:** Associate ID for test case ran
- **Tester:** Name of the team member that performed the test
- **Date/Time:** When the test case was performed
- **Result:** Description of what result occurred when the test was performed
- **Verdict:** Whether the test passed or failed

- **Defect ID:** If the test failed it will be given an associated defect ID and a new defect will be created.
- **Comments:** Any extra comments the team would like to include in the result.

9.4 Defects

A defect will be created if the test fails and will include the following:

- **Defect ID:** Unique ID number for Defect
- **Test Case Result ID:** Associated ID for test case result
- **Severity:** How important the defect is
- **Status:** Current status of test case
- **Error Log:** Information related to the test case such as error message or stack trace
- **Comments:** Any extra comments the team would like to include in the defect report

10. Test Schedule

We want to ensure that the project will perform exactly as specified in this document. To ensure that this is the case the team will produce several test cases that will test that each module functions in the expected way. These test cases will be expanded upon in the System Test Plan document.

10.1 MS Project Plan -- System Test Plan Phase

Table 10-1 System Test Phase Schedule

WBS	Task Name	Resource Names	Planned Start	Planned Finish	BCWS
3.5	**System Test Phase**		**Wed 4/1/15**	**Thu 5/7/15**	**49 days**
3.5.1	**System Test Phase I**		**Wed 4/1/15**	**Mon 4/20/15**	**15 days**
3.5.1.1	Unit tests for Presentation	Frank Robinson	Wed 4/1/15	Fri 4/17/15	1 day
3.5.1.2	Unit tests for Client Processing	Frank Robinson, Matthew Leonard	Wed 4/1/15	Fri 4/17/15	1 day
3.5.1.3	Unit tests for Data Storage	Matthew Leonard	Wed 4/1/15	Fri 4/17/15	2 days
3.5.1.4	Unit tests and hardware tests for Sensor Input	Dakota Slay	Wed 4/1/15	Sun 4/19/15	3 days
3.5.1.5	Unit tests and hardware tests for Drone Processing	Aaron Windham, Dakota Slay, Justin Crist	Wed 4/1/15	Sun 4/19/15	3 days
3.5.1.6	Unit tests and hardware tests for Hardware Output	Aaron Windham, Justin Crist	Wed 4/1/15	Sun 4/19/15	2 days

3.5.1.7	Modify parts and software	Aaron Windham, Matthew Leonard, Dakota Slay, Frank Robinson, Justin Crist	Fri 4/10/15	Mon 4/20/15	3 days
3.5.2	**System Test Phase II**		**Fri 4/17/15**	**Sat 5/2/15**	**19 days**
3.5.2.1	Component tests for Presentation	Frank Robinson, Matthew Leonard	Fri 4/17/15	Fri 4/24/15	3 days
3.5.2.2	Component tests for Client Processing	Frank Robinson, Matthew Leonard	Fri 4/17/15	Fri 4/24/15	3 days
3.5.2.3	Component tests for Data Storage	Frank Robinson, Matthew Leonard	Fri 4/17/15	Fri 4/24/15	3 days
3.5.2.4	Component tests for Sensor Input	Dakota Slay, Aaron Windham	Sun 4/19/15	Sun 4/26/15	3 days
3.5.2.5	Component tests for Drone Processing	Aaron Windham, Dakota Slay, Justin Crist	Sun 4/19/15	Sun 4/26/15	4 days
3.5.2.6	Component tests for Hardware Output	Aaron Windham, Justin Crist	Sun 4/19/15	Sun 4/26/15	3 days
3.5.2.7	Integration tests for client side	Frank Robinson, Matthew Leonard	Sun 4/19/15	Sat 5/2/15	3 days
3.5.2.8	Integration tests for drone side	Aaron Windham, Dakota Slay, Justin Crist	Sun 4/26/15	Sat 5/2/15	3 days
3.5.3	**System Test Phase III**		**Sat 5/2/15**	**Thu 5/7/15**	**15 days**
3.5.3.1	Full Integration of Client and Drone testing	Ground Control Team	Sat 5/2/15	Tue 5/5/15	5 days
3.5.3.2	System Validation Testing	Ground Control Team	Tue 5/5/15	Thu 5/7/15	10 days

11. Approval

Name	Role	Signature	Date
Mike O'Dell	Program Director		
Frank Robinson	Project Manager		
Justin Crist	Team Member		
Matthew Leonard	Team Member		
Dakota Slay	Team Member		
Aaron Windham	Team Member		
Paul Beaulieu	Project Sponsor		